

SISTEM PENGEMBANGAN KENDALI FUZZY LOGIC BERBASIS MIKROKONTROLER KELUARGA MCS51 (PetraFuz)

Thiang, Anies Hannawati, Ban Eng, Resmana

Fakultas Teknik, Jurusan Teknik Elektro - Universitas Kristen Petra

e-mail: thiang@petra.ac.id

ABSTRAK: Makalah ini menyajikan sebuah sistem pengembangan kendali fuzzy logic (PetraFuz, Petra Fuzzy Development System) yang dikembangkan oleh laboratorium Sistem Kontrol, Jurusan Teknik Elektro, Universitas Kristen Petra Surabaya. Sistem ini terdiri dari perangkat keras sistem mikrokontroler MCS51 dan perangkat lunak pendukung yang berjalan pada PC. Sistem PetraFuz digunakan untuk mengembangkan sistem berbasis fuzzy logic utamanya pada bidang kendali. Kemampuan sistem meliputi pengembangan pada fase perancangan kendali, evaluasi kendali, pembentukan program bahasa assembly MCS51 dan proses downloading program menuju target sistem mikrokontroler MCS51 untuk dieksekusi melakukan kendali pada plant yang nyata. Aksi kendali dapat diakuisi oleh program PC melalui komunikasi serial RS232 sehingga respon kendali dapat digambarkan pada layar monitor untuk dilakukan analisis lebih lanjut yang diperlukan pada proses tuning if-then fuzzy rules. Penggunaan tools PetraFuz, sangat membantu proses rancang bangun sistem kendali fuzzy terutama bagi praktisi/akademisi yang berminat menekuni teknik kendali yang relatif baru ini. Sistem PetraFuz diharapkan dapat memberikan kontribusi positif bagi masyarakat sistem kendali untuk mengembangkan berbagai aplikasi kendali berbasis fuzzy logic.

Kata kunci: logika fuzzy, sistem pengembangan kendali fuzzy, mikrokontroler MCS51

ABSTRACT : This paper presents a Fuzzy Logic Development Tool called PetraFuz which has been developed at Control System Laboratory, Electrical Engineering Department, Petra Christian University. The system consists of a hardware target based on MCS51 microcontroller and a software support running under PC Windows. The system is targeted for developing fuzzy logic based systems. It supports fuzzy logic design, evaluation, assembly language generator and downloading process to the target hardware to perform on-line fuzzy process. Process action and fuzzy parameters could be transferred to PC monitor via RS-232 serial communication, this on-line process parameters is used for fuzzy tuning, i.e. fuzzy if-then rules and fuzzy membership functions. The PetraFuz tool helps very much for Fuzzy system developments, it could reduce development time significantly. The tool could spur the development of fuzzy systems based on microcontroller systems such as fuzzy control systems, fuzzy information processing, etc.

Keywords : fuzzy logic, fuzzy control development system, microcontroller MCS51

1. PENDAHULUAN

Penggunaan teknik kendali Fuzzy Logic telah cukup meluas pada berbagai aplikasi mulai dari *consumer electronics*, *robotics*, kendali industri, dan lain-lain [1]. Implementasi kendali Fuzzy biasanya dilakukan oleh mikroprosesor multi-purpose, mikrokontroler maupun prosesor khusus berupa *Fuzzy Logic Processor*. Biasanya dibutuhkan alat atau software bantu untuk mengembangkan aplikasi fuzzy mulai dari tahap perancangan, evaluasi, implementasi dan penalaan (*tuning*).

Makalah ini menyajikan sebuah sistem pengembangan fuzzy logic (PetraFuz, Petra Fuzzy Development System) yang dikembangkan oleh laboratorium Sistem Kontrol, Jurusan Teknik Elektro, Universitas Kristen Petra Surabaya. Sistem ini terdiri dari perangkat keras sistem mikrokontroler MCS51 dan perangkat lunak pendukung yang berjalan pada PC. Pembuatan PetraFuz bertujuan untuk menyediakan alat/software bantu dalam pengembangan sistem berbasis fuzzy logic, utamanya pada aplikasi bidang kendali.

Proyek ini diinspirasi oleh proyek serupa yang dikembangkan oleh Motorola

(FUDGE, Fuzzy Design Generator) yang ditujukan untuk implementasi fuzzy pada mikroprosesor 68HC11, 6805 dan 68000 [2]. Sistem PetraFuz dikembangkan dengan kemampuan lebih yaitu berupa sistem pengembangan fuzzy yang terintegrasi dengan dilengkapi dengan sistem target perangkat keras. Program pada PC berinteraksi langsung dengan sistem target melalui komunikasi serial RS232. Sistem PetraFuz memilih platform mikrokontroler keluarga MCS51 karena ketersediaan dan kepopuleran penggunaan prosesor ini di Indonesia. Sistem PetraFuz diharapkan dapat memberikan kontribusi positif bagi masyarakat sistem kendali untuk mengembangkan berbagai aplikasi kendali berbasis *fuzzy logic*.

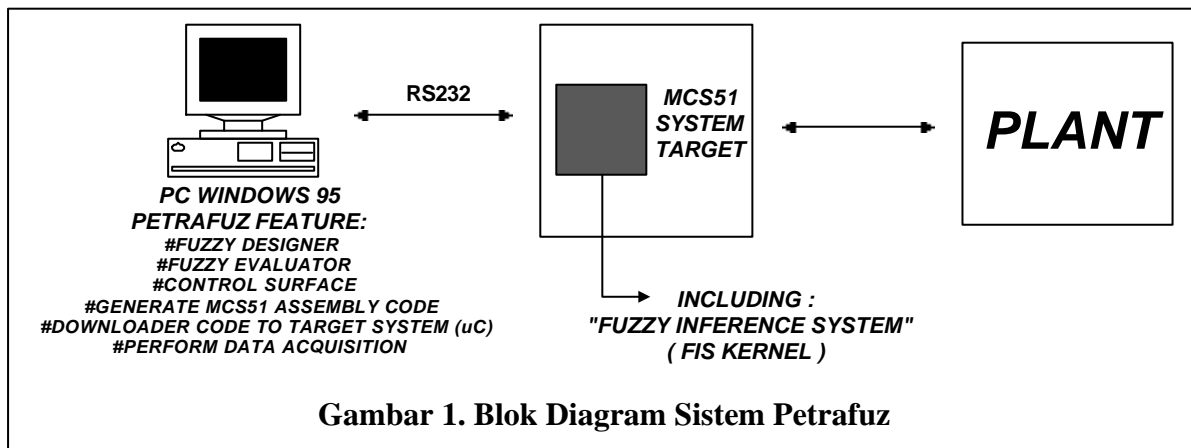
2. SISTEM PETRAFUZ

Sistem PetraFuz terdiri dari 2 bagian yaitu sistem software yang berjalan pada PC dan sistem target perangkat keras berupa sistem mikrokontroler MCS51. Blok diagram sistem PetraFuz seperti pada gambar 1.

dengan kemampuan *graphical user interface* sehingga memudahkan user dalam penggunaannya. Perancangan kendali meliputi pembentukan *fuzzy membership function* untuk input maupun output (maksimum 5 input, 3 output dan 8 label per input/output) serta pembentukan *fuzzy if-then rules*. Sedangkan proses evaluasi kendali menyajikan *control surface* untuk berbagai kombinasi input kendali. Dengan demikian pengguna dapat melihat respon kendali sebelum kendali sesungguhnya dijalankan oleh perangkat keras sistem target.

PetraFuz51 juga membentuk program bahasa assembly MCS51 yang selanjutnya di-*compile* dan di-*download* menuju sistem target melalui serial komunikasi RS232. Sistem target perangkat keras terdiri dari mikrokontroler MCS51 dilengkapi dengan paralel I/O serta *interface* analog (ADC/DAC).

Aksi kendali dapat diakuisisi oleh program PC melalui komunikasi serial RS232 sehingga respon kendali dapat digambarkan pada layar monitor untuk dilakukan analisis lebih lanjut yang diperlukan pada proses *tuning if-then fuzzy rules*. Selain respon kendali, masing-masing input *fuzzy logic*



Proses perancangan, evaluasi, pembentukan program assembly dan proses downloading MCS51 *machine code* menuju sistem target dilakukan oleh program yang berjalan pada PC yaitu PetraFuz51 software. Sedangkan sistem target melakukan proses fuzzy logic yang berinteraksi dengan perangkat I/O ke dunia luar. Proses fuzzy logic yang dilakukan oleh sistem target meliputi proses *fuzzification*, *rule evaluation* dan *defuzzification*. Program PetraFuz51 dibuat dengan Delphi v.2 berbasis Windows

juga dapat diamati melalui layar monitor.

Secara umum kemampuan sistem PetraFuz adalah:

- Max. 5 Input
- Max. 3 Output
- 8 Membership Functions per Input
- 8 Membership Functions per Output
- 1024 if-then Rule
- 15 Characters per Name (Input, Output, Member).
- 4 Points per Input Member. (Trapezoid MF)

- 1 Point per Output Member. (Singleton MF)

3. CONTOH PENGGUNAAN PETRA-FUZ

Program PetraFuz51 menyediakan beberapa fasilitas yaitu perancangan input dan output *membership function* (MF), perancangan *fuzzy if-then rules*, *fuzzy logic evaluator*, *control surface*, pembentukan *assembly code*, *downloader* ke sistem target dan *data acquisition*. Secara umum penggunaan PetraFuz dapat dibagi atas tahap-tahap berikut yaitu tahap pertama mendisain fuzzy inference system yang meliputi perancangan input dan output fuzzy beserta *membership function*-nya dan perancangan *fuzzy if-then rules*. Tahap kedua pembentukan database (yang terdiri dari MF dan if-then rules) dari hasil perancangan tahap pertama dalam bahasa assembly MCS51 yang akan digabungkan dengan *user program*. Adapun user program yang dimaksud adalah program yang dibuat user untuk akses data dari/ke I/O interface baik analog maupun digital sesuai dengan sistem kendali yang dirancang. Tahap ketiga adalah user program *dicompile* terlebih dahulu untuk menghasilkan *machine code*, baru kemudian *machine code* tersebut di-*download* ke sistem target mikrokontroler MCS51.

Untuk memberi gambaran yang lebih jelas diambil contoh penggunaan PetraFuz pada sistem kendali temperatur udara yang menggunakan bola lampu sebagai pemanas dan kipas sebagai pendingin. Input sistem fuzzy disini adalah *Error* dan *Delta_Error* temperatur udara terhadap *Setting Point* yang diinginkan. Input *Error* adalah selisih antara *Setting Point* dengan *Present Value* ($\text{Error} = \text{SP} - \text{PV}$) sedangkan *Delta_Error* adalah kecepatan perubahan error yang terjadi, $\text{Delta_Error} = \text{Error}(n) - \text{Error}(n-1)$.

Pertama dilakukan pengisian spesifikasi dari *crisp input* dan *crisp output* yaitu nama, satuan, nilai minimum, nilai maksimum dan jumlah label *membership function*. Tampilan menu dapat dilihat pada gambar 2 dan gambar 3.

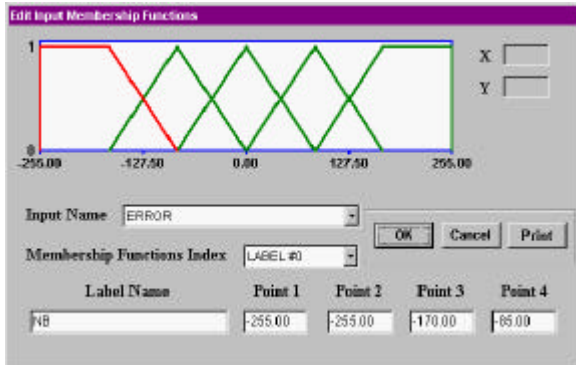
Gambar 2. Menu Edit Crisp Input

Gambar 3. Menu Edit Crisp Output

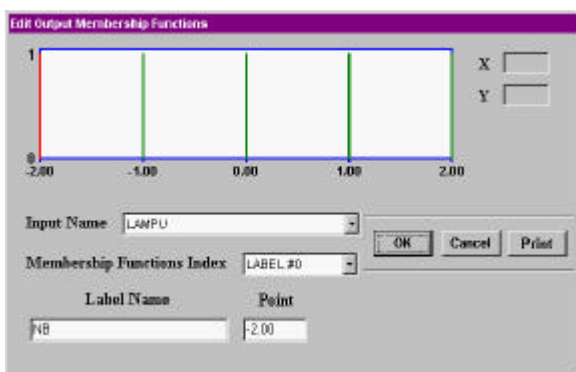
Dalam sistem kendali temperatur ini sebagai *crisp input* adalah *Error* dan *Delta_Error* sedangkan *crisp output* adalah Lampu dan Kipas. Dari gambar 2 dan 3 dapat dilihat bahwa input Error mempunyai 5 label *membership function* dengan nilai minimum -255 dan nilai maksimum 255 dan output lampu mempunyai 5 label *membership function* dengan nilai minimum -2 dan nilai maksimum 2. Input *Delta_Error* mempunyai 5 label dengan nilai minimum -255 dan nilai maksimum 255. Output kipas mempunyai 2 fuzzy label dengan nilai minimum -1 dan nilai maksimum 1.

Perancangan *membership function* dapat dilakukan dengan mudah baik secara grafis maupun dengan cara mengisi nilai point dari *membership function*. Bentuk input *membership function* hanya bisa dibuat dalam bentuk trapesium atau segitiga. *Membership function* ini dibentuk oleh 4 point yang dapat

dimasukan dalam program. Sedangkan output *membership function* hanya bisa dibuat dalam bentuk *singleton* yang dibentuk dengan 1 point yaitu posisi nilai *crisp* dimana MF ini berada. Tampilan pembentukan MF dapat dilihat pada gambar 4 dan gambar 5.



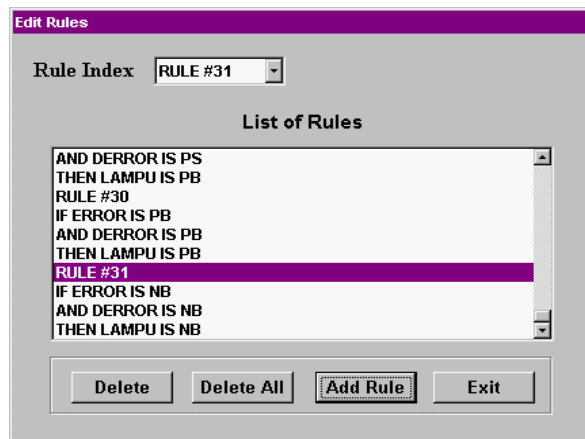
Gambar 4. Menu Input Membership Function



Gambar 5. Menu Output Membership Function

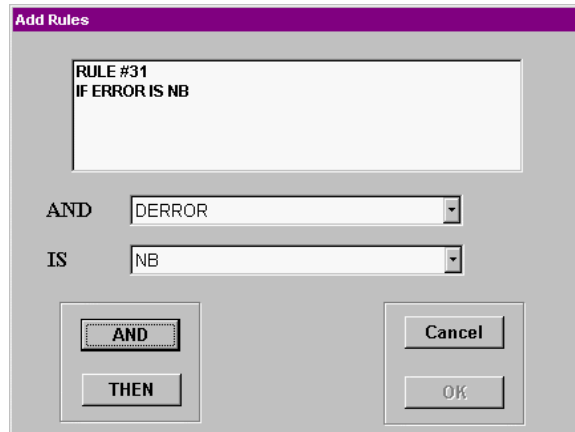
Dari gambar 4 dapat dilihat bahwa input *Error* mempunyai 5 label, 2 berbentuk trapesium, 3 berbentuk segitiga. Label 0 dari input *Error* diberi nama NB dengan nilai point1 = -255, point2 = -255, point3 = -170 dan point4 = -85 dan bentuk membershipnya adalah trapesium. Dari gambar 5 dapat dilihat bahwa output lampu mempunyai 5 label yang semuanya berbentuk single. Label 0 dari output lampu diberi nama NB dengan nilai point = -2 yang berarti mematikan 2 lampu. Nama label input *Error* adalah NB, NS, Z, PS, PB. Input *Delta_Error* mempunyai *membership function* sama dengan input *Error*. Output kipas hanya mempunyai 2 label yaitu N dan P.

Label N menyatakan kipas dimatikan, P menyatakan kipas dinyalakan.



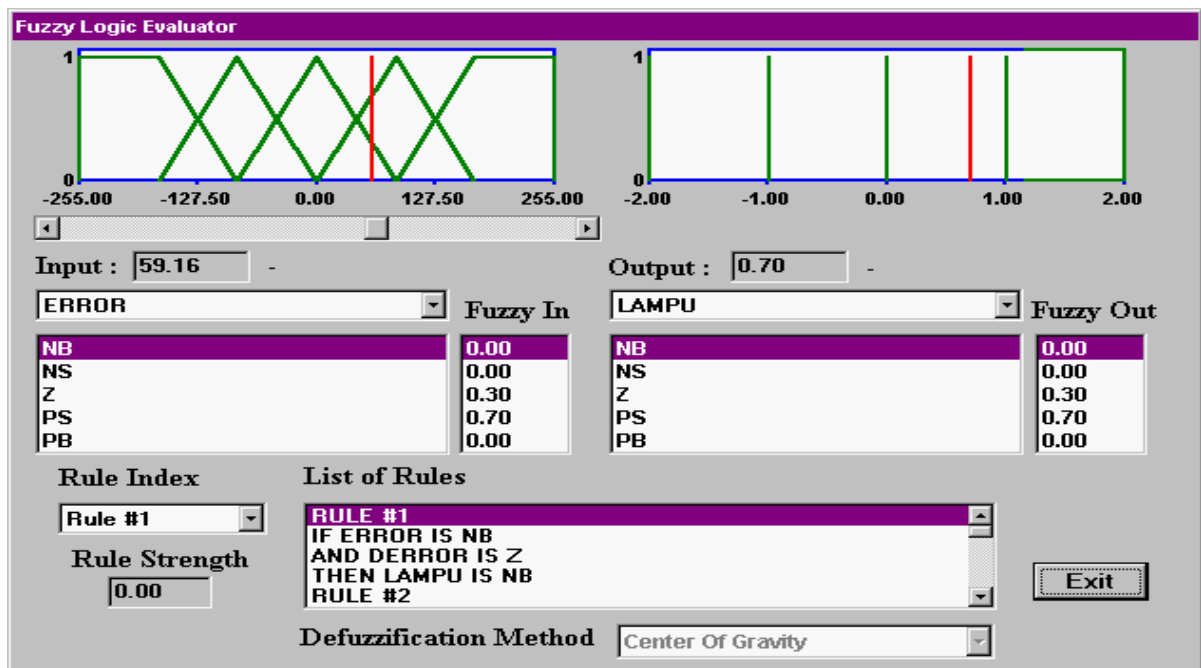
Gambar 6. Menu Edit Rule

Gambar 6 menampilkan tampilan menu pembentukan *fuzzy if-then rules*. Kapasitas maksimum rule yang dapat dibuat dalam program PetraFuz51 adalah sebanyak 1024 rule dengan tanpa membatasi jumlah *antecedent* dan *consequent* dari tiap rule. Untuk menambah rule dapat dilakukan dengan *click* pada *Add Rule* dan akan keluar tampilan seperti gambar 7.



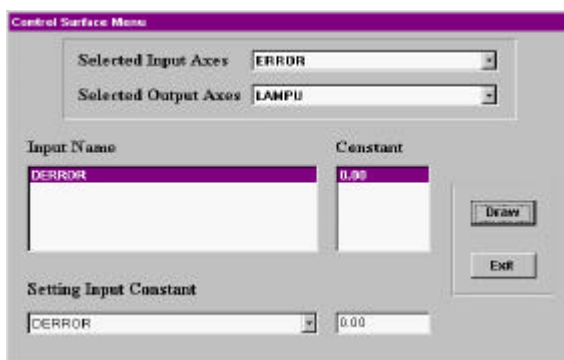
Gambar 7. Menu Add Rule

Fasilitas *fuzzy logic evaluator* disediakan untuk evaluasi sistem kendali yang dirancang. Pengguna dapat mengetahui hasil fuzzifikasi, hasil evaluasi rule yaitu rule-rule mana yang aktif, hasil defuzzifikasi bila input diset pada nilai tertentu. Dengan demikian memberikan gambaran kepada pengguna tentang aksi kendali fuzzy logic yang telah dirancang. Tampilan menu dapat dilihat pada gambar 8.



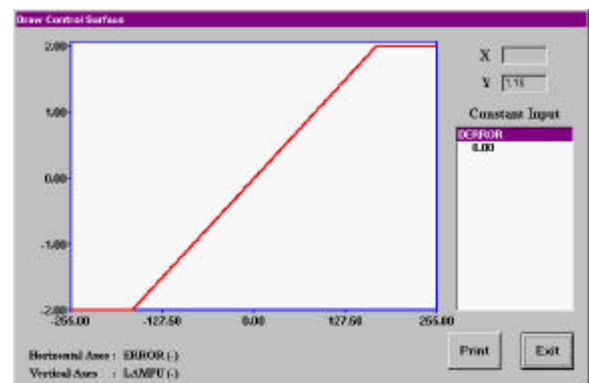
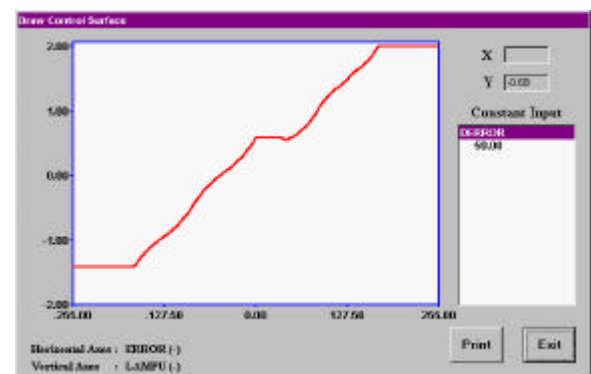
Gambar 8. Tampilan Menu Fuzzy Logic Evaluator

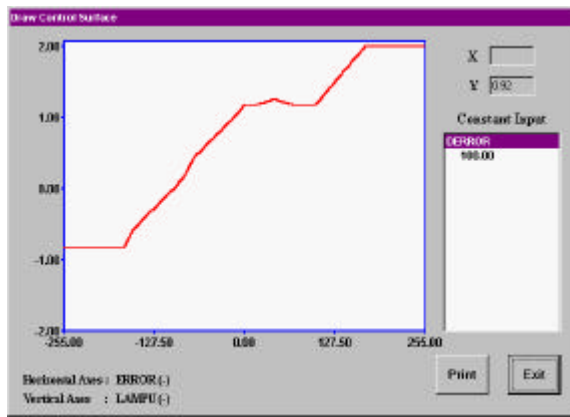
Fasilitas *control surface* disediakan untuk mengevaluasi aksi sistem kendali yang dirancang sebelum sistem kendali yang sesungguhnya di jalankan oleh sistem target. *Control surface* menyajikan grafik fuzzy output terhadap variasi salah satu input. Bila input lebih dari satu maka input yang lainnya dianggap konstan. Tampilan menu *control surface* dapat dilihat pada gambar 9. Disini ditampilkan grafik output Lampu terhadap input *Error* dengan nilai *Delta_Error*=0. Grafik dapat dibentuk untuk nilai *Delta_Error* lainnya yang dikehendaki.



Gambar 9. Tampilan Menu Control Surface

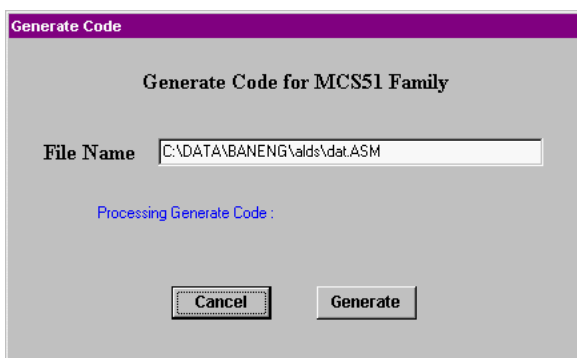
Grafik output lampu terhadap input *Error* sistem kendali temperatur yang dirancang dapat dilihat pada gambar 10, 11 dan 12.

Gambar 10. Grafik Output Lampu terhadap Input Error dengan $\Delta_Error = 0$ Gambar 11. Grafik Output Lampu terhadap Input Error dengan $\Delta_Error = 50$



Gambar 12. Grafik Output Lampu terhadap Input Error dengan Delta_Error = 100

Membership function dan *rule* yang telah didesain harus dikonversi ke dalam bentuk bahasa *assembly* MCS51 agar bisa digabungkan dengan *user program* yang akan di-*download* ke sistem target mikrokontroler MCS51. Hal ini dapat dilakukan dengan memanfaatkan fasilitas *generate code*. Fasilitas ini akan mempersiapkan database dari *membership function* dan *rule* dan menyimpannya dalam file dengan extension *.ASM*. Semua range nilai baik untuk input maupun output akan dikonversi (normalisasi) ke range 0 – FFh sehingga dalam program bahasa *assembly* dapat diminimalkan penggunaan bilangan negatif. Tampilan menu *generate code* dapat dilihat pada gambar 13.

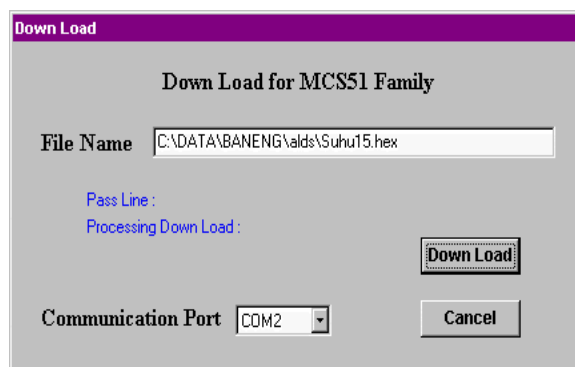


Gambar 13. Menu Generate Code

File *.ASM* yang dihasilkan mempunyai format khusus, hanya bisa digunakan untuk PetraFuz Routine Engine yang tersedia dalam kernel sistem target mikrokontroler MCS51. PetraFuz Routine Engine adalah

suatu procedure yang melakukan proses fuzzy logic meliputi proses fuzzifikasi, pengevaluasian rule dan defuzzifikasi. Metode defuzzifikasi yang digunakan adalah *Center of Gravity* atau *Center of Area*. Dengan demikian user hanya perlu membuat program untuk interaksi dengan I/O ke dunia luar. Data yang dibaca dimasukkan ke parameter input fuzzy, PetraFuz routine yang akan melakukan proses fuzzy logic dan menghasilkan output fuzzy, yang oleh user program akan diolah dan dikeluarkan ke *interface* dengan dunia luar. Routine ini bersifat *general*, berlaku untuk jumlah input dan output yang bervariasi (maksimum 5 input dan 3 output) serta jumlah *rule* yang bervariasi (maksimum 1024 *rule*).

File hasil bentukan *Generate Code* selanjutnya digabungkan dengan *user program* lalu di-*compile*. Setelah *user program* di-*compile* membentuk *machine code*, maka *machine code* tersebut selanjutnya di-*download* ke sistem target mikrokontroler sehingga sistem mikrokontroler dapat menjalankan proses kontrol fuzzy logic. *User program* harus di-*compile* dalam bentuk *Intel Hex Format* karena program *downloader* yang dibuat dalam program PetraFuz51 disini bekerja berdasarkan format *Intel Hex*. Tampilan menu *downloader* dapat dilihat pada gambar 14.

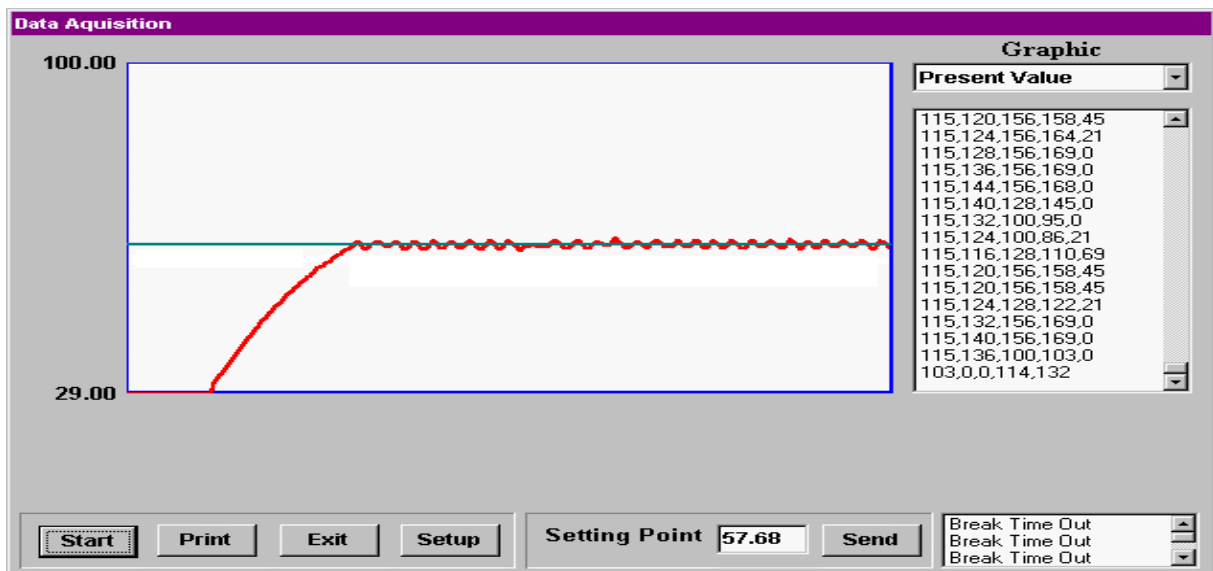


Gambar 14. Menu Downloader

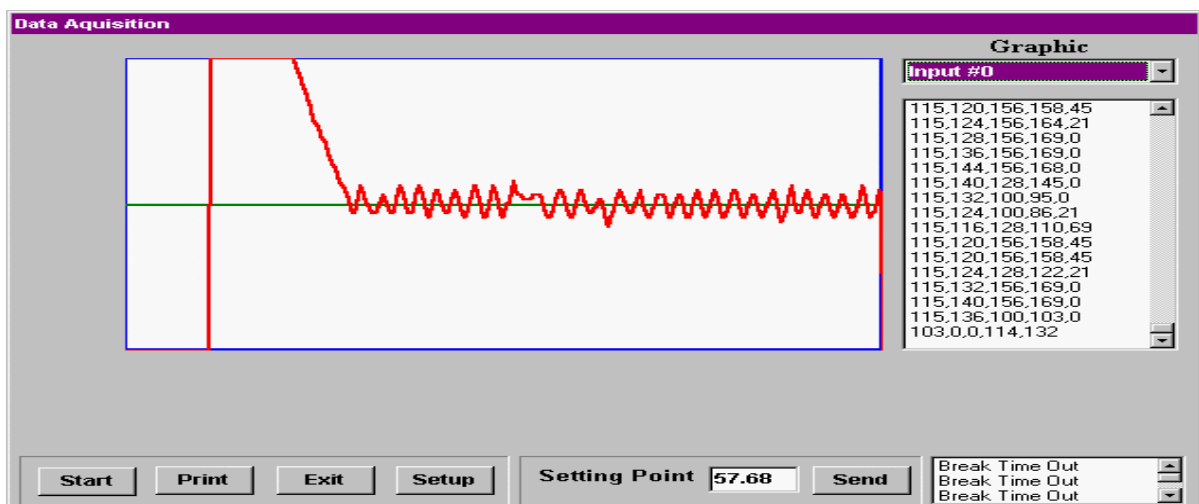
Fasilitas *data acquisition* disediakan untuk akuisisi data dari sistem target mikrokontroler. Pengguna dapat melihat respon sistem kendali dan juga perubahan nilai semua input dan output terhadap waktu. Dengan demikian pengguna dapat meng-

evaluasi respon sistem kendali yang sedang dijalankan oleh sistem target secara online. Kemampuan ini sangat dibutuhkan terutama dalam proses penalaan MF dan if-then rules. Proses perancangan bisa diperbaharui dan kembali di-*download* ke sistem target untuk dilihat responnya. Proses ini dapat dilakukan secara berulang dengan mudah untuk

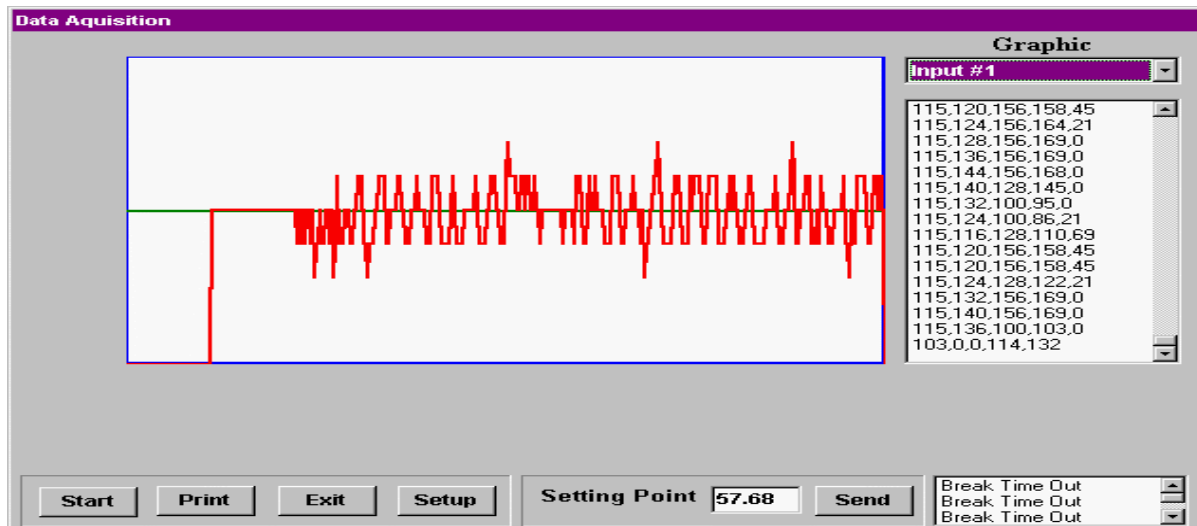
mendapatkan respon kendali yang optimum. Berikut ini adalah gambar dari hasil akuisisi data sistem kendali temperatur udara yang dirancang. Disini terlihat respon kendali terdapat error terhadap setting point yang diinginkan dan membutuhkan penalaan lebih lanjut. Tampilan dari *data acquisition* ini dapat dilihat pada gambar 15,16 dan 17.



Gambar 15. Grafik Nilai Respon Sistem



Gambar 16. Grafik Input Error



Gambar 17. Grafik Input Delta_Error

```

.CODE
ORG 2100H
INPUT_MFS      ;Input Membership Functions
IN0MF          ;ERROR
               DB 000H,0FFH,02AH,006H ;NB
               DB 02AH,006H,055H,006H ;NS
               DB 055H,006H,080H,006H ;Z
               DB 080H,006H,0AAH,006H ;PS
               DB 0AAH,006H,0FFH,0FFH ;PB
.CODE
ORG 21A0H
OUTPUT_MFS     ;Output Membership Functions
OUT0MF         ;LAMPU
               DB 000H ;NB
               DB 040H ;NS
               DB 080H ;Z
               DB 0BFH ;PS
               DB 0FFH ;PB
.CODE
ORG 21C8H
RULE_START     DB 000H ;IF ERROR IS NB
               DB 00AH ;AND DERROR IS Z
               DB 080H ;THEN LAMPU IS NB

               DB 000H ;IF ERROR IS NB
               DB 00BH ;AND DERROR IS PS
               DB 081H ;THEN LAMPU IS NS

```

Gambar 18. Potongan Struktur Database Membership Function dan Rule

4. PROGRAM KERNEL FUZZY INFERENCE SYSTEM

PetraFuz Routine Engine adalah *routine* untuk melakukan proses *fuzzy inference*. *Routine* ini tersedia pada kernel sistem target mikrokontroler MCS51. *Routine* ini didesain khusus untuk format database yang di-generate oleh *PetraFuz51 software*. Gambar

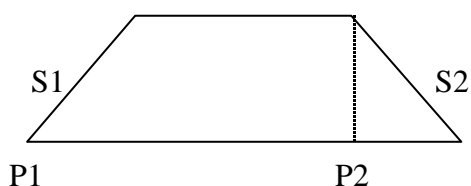
18 adalah potongan database hasil *generate* *PetraFuz*.

Struktur database untuk input *membership function* untuk setiap label menggunakan memory sebesar empat byte. Byte pertama dan ketiga adalah nilai *point* 1 (P1) dan *point* 2 (P2) sedangkan byte kedua dan keempat adalah nilai *slope* 1 (S1) dan *slope* 2 (S2). Untuk lebih jelas posisi dari

P1, P2, S1 dan S2 dapat dilihat pada gambar

Struktur database untuk output *membership function* hanya memerlukan memory satu byte yaitu untuk nilai *point* masing-masing label dari output *membership function*. Sedangkan struktur database dari *rule* bergantung pada jumlah *antecedent* dan jumlah *consequent* untuk tiap-tiap *rule*. Untuk contoh di atas dimana setiap *rule* terdapat dua *antecedent* dan satu *consequent* maka besar memory yang digunakan adalah tiga byte untuk setiap *rule*.

PetraFuz *routine* terletak dalam internal ROM pada address 040Ah. *Address crisp input* adalah 08h-0Ch sedangkan *address untuk crisp output* adalah 0Dh-0Fh. *Address* 08h untuk *crisp input* 1, *address* 09h untuk *crisp input* 2, *address* 0Ah untuk *crisp input* 3 dan seterusnya. Demikian juga untuk *crisp output*, *address* 0Dh untuk *crisp output* 1, *address* 0Eh untuk *crisp output* 2 dan *address* 0Fh untuk *crisp output* 3. Jadi untuk menjalankan PetraFuz *routine* untuk melakukan proses *fuzzy inference* dapat dilakukan dengan *call* pada alamat 040Ah. Tentunya sebelum dijalankan, *user program* harus menyimpan nilai-nilai *crisp input* ke *address* yang telah ditentukan, demikian juga untuk mengambil hasil dari proses *fuzzy inference* dapat dibaca dari *address* yang telah disebutkan di atas.



Gambar 19. Struktur Database Input Membership Function

5. PELUANG PENGGUNAAN PETRA-FUZ

Selama ini cukup banyak para praktisi sistem kendali mengalami kesulitan dalam mengaplikasikan metode fuzzy logic dalam sistem kendali terutama sampai pada level implementasi hardware. Salah satu alasannya adalah karena tidak didukung oleh *fuzzy*

19.

development tools yang dilengkapi dengan target sistem perangkat keras. Sistem PetraFuz menyediakan tools yang cukup lengkap mulai dari proses disain *membership function* dan pembentukan *fuzzy if-then rule* sampai pada proses kendali fuzzy logic yang dilaksanakan oleh sistem target perangkat keras. Pada kenyataannya dalam kendali fuzzy logic, bagian yang tersulit adalah mendisain *membership function* dan *fuzzy rules* sehingga didapatkan yang optimum dan menghasilkan kendali fuzzy logic yang memadai. Untuk mendapatkan hasil yang terbaik maka harus dilakukan percobaan yang berulang-ulang. Ini akan menjadi suatu kendala tersendiri bila tidak mempunyai *tools* yang menunjang.

Dari hasil pengujian yang dilakukan, PetraFuz *Routine Engine* mampu mengevaluasi lebih kurang 16000 rule dalam satu detik. Pengujian ini dilakukan dengan kondisi tiap rule mempunyai 2 *antecedent* dan 1 *consequent* dan sistem *clock* yang digunakan pada prosesor MCS51 adalah 2 MHz. Ini berarti untuk mengevaluasi satu rule, PetraFuz *Routine Engine* membutuhkan waktu lebih kurang 62,5 mikrodetik. Jika sistem kendali fuzzy logic mempunyai 2 input 1 output dan masing-masing input mempunyai 5 label *membership function* maka jumlah maksimum rule adalah 25 sehingga waktu untuk mengevaluasi rule tidak lebih dari 1,5625 milidetik.

Dengan waktu akses yang cukup cepat ini, maka sistem PetraFuz dapat digunakan pada sistem kendali yang mempunyai respon yang relatif cepat seperti mengendalikan kecepatan motor DC, kontrol posisi dan lain-lain. Selain itu sistem ini berpeluang untuk digunakan pada aplikasi proses kendali industri seperti *pressure control*, *chemical process control*, *kiln control* dan lain-lain.

Sistem ini juga berpeluang dimanfaatkan untuk pengembangan kendali pada berbagai bidang. Misalkan untuk bidang pertanian, penggunaannya pada kendali temperatur untuk *cold/hot storage* produk-produk pertanian yang membutuhkan kondisi temperatur tertentu. Dapat pula digunakan untuk pengendalian kadar PH air tambak, identifi-

kasi level kematangan produk pertanian, dan

Sistem PetraFuz telah dimanfaatkan sebagai modul praktikum Kendali Fuzzy Logic pada institusi penulis dan terlihat cukup bermanfaat bagi pemula untuk memahami dan merancang sistem kendali fuzzy. Disamping itu pula *tools* ini memberikan pemahaman aspek implementasi khususnya menggunakan mikrokontroler.

Penulis juga telah mengembangkan *fuzzy routine/kernel* program untuk jenis mikroprosesor 8088 yang telah dimanfaatkan untuk aplikasi prediksi jumlah kendaraan bermotor berdasarkan level kebisingan/*noise* lalu lintas.

Program PetraFuz dapat digunakan secara gratis dan dapat di-*download* pada alamat web Internet <http://faculty.petra.ac.id/resmana/basiclab/petrafuz.htm>

6. KESIMPULAN

Dari hasil eksperimen pembuatan dan pemanfaatan sistem pengembangan fuzzy logic PetraFuz penulis mencatat beberapa hal antara lain:

- Sistem PetraFuz relatif mudah digunakan dan bermanfaat bagi pemula yang ingin bereksperimen sistem kendali berbasis fuzzy logic yang diimplementasikan pada mikrokontroler. Saat ini mikrokontroler yang dapat digunakan terbatas pada keluarga MCS51 serta 8088, dan tidak tertutup kemungkinan untuk dikembangkan fuzzy kernel untuk berbagai prosesor misalkan MCS96, Z80 atau lainnya.
- Sistem PetraFuz menyediakan kemampuan yang interaktif dan terintegrasi dengan sistem target perangkat keras. Dengan *tools* ini meningkatkan peluang pemanfaatan konsep kendali fuzzy pada berbagai bidang di Indonesia, khususnya pada bidang kendali.
- Masih terbuka pengembangan lebih lanjut dari sistem ini untuk mengakomodasi jumlah input/output yang lebih banyak, variasi bentuk *membership function* dan berbagai metode defuzzifikasi. Juga tidak menutup kemungkinan untuk memadukan metode jaringan saraf tiruan dengan

lain-lain.

fuzzy logic untuk membentuk kendali cerdas.

UCAPAN TERIMA KASIH

Pada kesempatan ini penulis ingin mengucapkan terima kasih kepada asisten laboratorium Sistem Kontrol jurusan Teknik Elektro Universitas Kristen Petra yaitu Jaury Adi Wijaya, Silviya, Kardy Antolis, Darwin, Eric PS atas dukungan dan kerja samanya selama proyek ini dikerjakan sehingga proyek ini bisa diselesaikan. Tak lupa juga ucapan terima kasih kepada laboran laboratorium Sistem Kontrol, bapak Heri Soehartono atas dukungan dan bantuan yang diberikan selama pengerjaan proyek ini. Serta terima kasih kami tujukan kepada teman-teman dosen Jurusan Teknik Elektro atas dukungan membangun sehingga proyek ini terlaksana.

DAFTAR PUSTAKA

1. Klir, George J, *Fuzzy Sets and Fuzzy Logics : Theory and Applications*, NJ : Prentice Hall, 1995.
2. Alex De Castro, Jason Spelman, John Dumas, *Fuzzy Designer Generator (FUDGE)*, Motorola Inc., 1994
3. Center for Emerging Computer Technologies, *Fuzzy Logic Education Program*, Motorola Inc., 1994
4. Advanced Micro Device, *Microcontrollers Handbook*, California : Advanced Micro Device, 1988.